

Discussion Paper / Artículo de Reflexión - Tipo 2

Inference model for dynamic classification of monographs at university level

Germán Alejandro Pinilla García / gapinillag@correo.udistrital.edu.co

Julio Barón Velandia, MSc. / jbaron@udistrital.edu.co

Universidad Distrital Francisco José de Caldas, Bogotá-Colombia

ABSTRACT The inappropriate visibility of degree documents in monograph mode prevents new members of the academic community from querying and accessing them, significantly limiting the possibility of giving continuity and complementing versions of already completed work, such as applying the results both industrial and socially. This article proposes a knowledge base model supported on ontology to improve the relevance of documents in the digital level as a result of a search on specific interest aspects. To produce new knowledge, predicate logic is used and applied to the ontological model of the representation of monographs, in order to establish dynamic and reliable inferences or to make changes in situations where required.

KEYWORDS Inference, knowledge base, monograph, ontology, predicate logic.

Modelo de inferencia para clasificación dinámica de monografías a nivel universitario

RESUMEN La inadecuada visibilidad de documentos de grado en modalidad monografía impide la consulta y acceso por parte de nuevos integrantes de la comunidad académica, limitando significativamente la posibilidad de dar continuidad y complementar versiones de trabajos ya finalizados, así como la aplicación de resultados tanto a nivel social como industrial. Este artículo propone un modelo de base de conocimiento soportado en una ontología para mejorar la pertinencia de documentos presentados a nivel digital como resultado de una búsqueda sobre aspectos de interés específico. Para la obtención de nuevo conocimiento se hace uso de la lógica de predicados aplicada sobre el modelo ontológico de representación de monografías, con el fin de establecer de manera dinámica y confiable inferencias o llegar a realizar modificaciones en las situaciones donde sean requeridas.

PALABRAS CLAVE Base de conocimiento, inferencia, lógica de predicados, monografía, ontología.

Modelo de inferência para a classificação dinâmica de monografias de nível superior

RESUMO A inadequada visibilidade de trabalhos de grau na modalidade de monografia impede a consulta e acesso de novos membros da comunidade acadêmica, limitando significativamente a possibilidade de continuar e complementar as versões dos trabalhos já concluídos, bem como a aplicação dos resultados a nível social e industrial. Este artigo propõe um modelo de base de conhecimento suportado por uma ontologia para melhorar a relevância de trabalhos apresentados digitalmente, como resultado de uma pesquisa sobre aspectos de interesse específico. Para obter um novo conhecimento utiliza-se a lógica de predicados aplicada ao modelo ontológico de representação de monografias, a fim de estabelecer inferências de maneira dinâmica e fiável ou fazer alterações em situações em que sejam necessárias.

PALAVRAS-CHAVE Base de conhecimento; inferência; lógica de predicados; monografia; ontologia.

I. Introduction

The techniques of information management have been very important from the second half of the twentieth century. There have thus emerged data structures called databases, in the mid-Fifties, a technique used in the majority of universities. As the name suggests, these tools were designed to facilitate the storage of huge amounts of sorted data in a record layout, allowing querying and modifying in an agile and secure way. The databases are composed of four main components: data, the programs, storage devices, and users. Besides, the integration, and redundancy removal in the management of data files are its main advantage (Alonso, 1992). Examples of these tools include international systems of thesis classification such as Teseo (2015) and Cybertesis (2015); and national level implementations at the Universidad Nacional de Colombia (Sinab, 2015) and at the Universidad Libre (Repositorio Unilibre, 2015).

In addition to the systematic recording of information, the incorporation of artificial intelligence techniques into the traditional models of databases has enabled the storage of knowledge elements expressed by facts, rules, and systems to determine and modify the relationships between facts (Beekman, 1999). This proposal is known as a knowledge base and is defined in a synthetic way as an integration between a database and artificial intelligence (Wang, 2011). The definition most suited to the proposal of this Project is one which describes a representation model of knowledge composed of three parts, logical, ontological and computational (Hao, 2011) and this model represents a world, company or section of reality (Mylopoulos & Levesque 1983).

A knowledge base, unlike the databases that allow the querying and modifying of the facts of a theme or specific domain, also allows storage rules that use the collection of existing facts to extract information not stored in an explicit way. Each specific theme is represented by a collection of entities or elements, and a collection of relationships between these. These collections create states that can generate transformations in the creation of elements and modifications of the relationships between these. The progress in the scientific research on the correction of the knowledge base has been significant, in contrast to the gap in the search for techniques for the dynamic verification of the rules in the inference process (Yoo & Park, 1993). Therefore, the rules are reviewed and inferences are performed at the moment of the user's request to the database.

I. Introducción

La búsqueda de técnicas de gestión de la información ha tomado gran importancia a nivel computacional desde la segunda mitad del siglo XX. Es así como a mediados de los 50 surgen las estructuras llamadas bases de datos, una técnica que se usa en la mayoría de universidades. Estas herramientas pretendían como sugiere su nombre, facilitar el almacenamiento de grandes volúmenes de datos ordenados obedeciendo esquemas, para luego poder ser recuperados y/o modificados de una manera ágil y segura. Las bases de datos están formadas por cuatro componentes principales: datos, programas, dispositivos de almacenamiento y usuarios, y su principal ventaja en el manejo de ficheros de datos es la integración y eliminación de redundancias entre ellos (Alonso, 1992). Ejemplos de esta herramienta se encuentran a nivel internacional en los sistemas organizadores de tesis como Teseo (2015) y Cybertesis (2015), y a nivel nacional con implementaciones como la de la Universidad Nacional de Colombia (Sinab, 2015) y la propuesta utilizada por la Universidad Libre (Repositorio Unilibre, 2015).

La incorporación de técnicas de inteligencia artificial a los modelos convencionales de bases de datos ha permitido, además del registro sistemático de información, el almacenamiento de elementos de conocimiento expresados mediante hechos, reglas y sistemas para determinar y modificar las relaciones existentes entre los hechos (Beekman, 1999); esta propuesta es conocida como base de conocimiento y se define, de manera sintética, como una integración entre una base de datos y la inteligencia artificial (Wang, 2011). La definición que más se ajusta a la propuesta de este proyecto es la que las considera como un modelo de un mundo, empresa o sección de la realidad (Mylopoulos & Levesque 1983) que propone un modelo de representación de conocimiento compuesto de tres partes: lógica, ontológica y computacional (Hao, 2011).

La principal diferencia entre una base de datos y una base de conocimientos radica en que la segunda, además de permitir la consulta y modificación de los hechos de un tema o dominio específico, almacena reglas que utiliza el conjunto de hechos existentes para extraer información que no está registrada de manera explícita, representando cada tema específico mediante una colección de entidades o elementos y una colección de relaciones existentes entre las entidades, creando estados que pueden tener transformaciones en la creación de elementos o modificaciones de relaciones entre ellos. El avance investigativo en la corrección de bases de conocimiento ha sido considerable, pero ha existido un retraso en la búsqueda de técnicas para verificar las reglas de manera dinámica durante el proceso de inferencia (Yoo & Park, 1993). Por consiguiente, las reglas son revisadas y se realizan inferencias en el momento en que el usuario consulta la base de conocimiento.

Como apoyo a estas técnicas, la ingeniería de conocimiento se ha encargado de crear esquemas de representación del conocimiento, tales como grafos conceptuales y redes semánticas,

los cuales hacen uso de conceptos como la herencia, buscando obtener modelos cada vez más exactos y aproximados a la realidad. Entre estos modelos también se encuentran las ontologías, las cuales permiten representar un conjunto de conceptos organizados de un tema específico y hacer inferencias, como el proyecto Cyc (Citycorp, 2015) que busca realizar una integración entre una ontología y una base de conocimiento utilizada para realizar inferencias (Lenat & Guha, 1990). El uso de ontologías permite usar símbolos primitivos para representar significados, establecer relaciones semánticas entre conceptos, y solucionar problemas de ambigüedades semánticas en cuanto a la interpretación, y constituye una buena herramienta de clasificación de elementos tales como: personas, lugares y roles, entre los que pueden ser realizadas inferencias para obtener información no explícita en el conocimiento representado, mediante lógicas descriptivas –como la lógica de predicados–, usando diferentes elementos para hacer verificaciones y validaciones de hechos que se encuentren en la base de conocimiento modelada.

Este trabajo presenta una propuesta de una base de conocimiento enfocada en monografías, basada en una ontología como recurso para representación y especificación terminológica del dominio. En la propuesta se define monografía como “una investigación académica elaborada por un estudiante, en un tema elegido por él mismo, bajo la dirección de un profesor o investigador experto en el área específica de conocimiento a la que se refiere la investigación” (Universidad del Rosario, 2011, p.1). La representación de la información, mediante una ontología de dominio específico, permite representar conocimiento especializado (Gangemi, Pisanelli, & Steve, 1998), para obtener conocimiento que no está representado de manera explícita mediante el uso de la lógica de predicados a través de inferencias, proporcionando nuevas formas de visibilizar los trabajos de monografía desarrollados con anterioridad en la instituciones universitarias, para que puedan servir de referencia y base para nuevos trabajos.

Este documento está conformado de la siguiente manera: la sección 2 presenta los fundamentos conceptuales que soportan la propuesta; la sección 3 presenta la metodología utilizada para su desarrollo, inicia con un fragmento de la ontología que ilustra las características generales de los trabajos de grado sobre la cual se aplicó la lógica de predicados de primer orden, continúa con la equivalencia de la ontología en el lenguaje computacional OWL, y finaliza con el planteamiento de la lógica de predicados expresada en notación algebraica, para luego obtener las inferencias; la sección 4 presenta el resultado de las inferencias obtenidas para mejorar las actividades en las universidades; y las secciones 5 y 6 presentan las conclusiones, recomendaciones y el trabajo a futuro.

II. Fundamentos conceptuales

A. Base de conocimiento

Herramienta que permite almacenar y facilita el acceso al

In support of these techniques, knowledge engineering has created schemes for knowledge representation such as conceptual graphs and semantic networks. These schemes seek to obtain models that are more exact and closer to reality, using concepts such as inheritance. Ontologies are other models that allow the representation of an organized collection of concepts on a specific theme and inferences to be made based on these concepts. An example of an ontology is the Cyc project (Citycorp, 2015), the aim of which is to perform an integration between an ontology and a knowledge base used to execute inferences (Lenat & Guha, 1990). The use of primitive symbols in the ontologies allows us to represent meanings, establish semantic relationships between concepts, and solve semantic problems of ambiguity in interpretation. This constitutes a good classification tool for elements such as: people, places and roles, and it is possible to make inferences based on these elements to obtain information that is not explicit in the knowledge represented. This is performed through descriptive logic such as the logic of predicates, using different elements for the verification and validation of facts that are contained in the knowledge base modeled.

This paper presents a proposed knowledge base focused on monographs and based on an ontology as the resource for the representation and terminological specification of the domain. In this proposal the monograph concept is defined as “an academic research prepared by a student, on a specific theme chosen by himself, under the direction of a teacher or researcher who is expert in the specific knowledge area of the research” (Universidad del Rosario, 2011, p. 1). Information representation by a specific domain ontology enables to represent specialized knowledge (Gangemi, Pisanelli, & Steve, 1998). With this representation and using the predicate logic through inferences, it is possible to obtain knowledge that is not present in an explicit way. Finally, this provides a new manner of visualizing the monographs developed at universities that can be referenced from other theses.

This document is composed as follows: section 2 presents the conceptual framework that supports the proposal; section 3 presents the methodology used for this development, beginning with a fragment of the ontology which illustrates general features of the monographs and on which is applied the predicate logic of first order; continues with the ontological equivalence in the OWL computational language, and concludes with the proposed predicate logic expressed in algebraic notation to achieve the inferences; section 4 pre-

sents the results from the inferences to improve activities at universities; sections 5 and 6 present the conclusions, with recommendations and future work.

II. Conceptual Framework

A. Knowledge base

This is a tool that allows storage and facilitates access to knowledge, which is all the information generated and stored in an organization with determinate objectives for its use. The knowledge can be explicit -when it is storable and retrievable - or tacit -when experiences cannot be encoded (Díez & Zúñiga, 2011). In the field of artificial intelligence, these bases are defined as a “persistent collection of knowledge that supports the operation of a smart system” (Holmes & Stocking, 2009, p.3).

A knowledge base is composed of an ontology, as well as the facts and rules expressed through a declarative language; this latter contains the conceptual terminology only, while the knowledge base contains both information and facts. This concept has been widely applied in digital libraries, databases, and expert systems (Lahaba & León, 2001). Although these are deficient at the organizational level at the moment for representing the knowledge that can be generated in a consistent and dynamic way, the concept is also used in thesaurus and traditional systems of classification.

A solution to the problem of reuse of knowledge, which emerged during the last decade of the twentieth century, was established as a basis for the creation of research projects with the objective of achieving improvements in the knowledge representation. In this way, a knowledge management system searches for the best manner of organizing, controlling, and providing the existing knowledge about some area, so that it can be used.

B. Ontologies

Ontologies are a flexible tool in knowledge representation. Hendler (2001) defines them as “a collection of knowledge terms that include a vocabulary, relationships, and a set of logical rules and of inference about a particular domain”. Meanwhile, computer science defines them as “an explicit specification of a conceptualization” (Gruber, 1995, p. 2).

Ontologies are composed of elements such as the classes or concepts from a domain that are properties represented using relationships and attributes; as well as of individuals or instances of the same classes and axioms,

conocimiento, es decir a toda información que es generada y almacenada en una entidad, para ser usada de acuerdo con determinados objetivos; el conocimiento puede ser explícito –cuando es almacenable y recuperable– o tácito –cuando corresponde a experiencias y vivencias que no pueden ser codificadas– (Díez & Zúñiga, 2011). En el campo de la inteligencia artificial estas bases se definen como una “colección persistente de conocimiento que soporta la operación de un sistema inteligente” (Holmes & Stocking, 2009, p.3).

Una base de conocimiento está compuesta por una ontología, así como los hechos y reglas expresadas mediante un lenguaje declarativo; esta última contiene únicamente la terminología conceptual, mientras que la base de conocimiento contiene, tanto la información, como los hechos. Este concepto ha sido ampliamente aplicado en bibliotecas digitales, bases de datos y sistemas expertos (Lahaba & León, 2001). Aunque presentan deficiencias a nivel organizacional al momento de representar el conocimiento que puede ser generado de forma consistente y dinámica, también es utilizada en tesauros y sistemas tradicionales de clasificación.

Como solución al problema de reutilización de conocimiento, que surgió durante la última década del siglo XX, se constituyó en la base para la creación de proyectos de investigación que pretendían buscar mejoras en la representación del conocimiento. De esta manera, un sistema de gestión de conocimiento busca la mejor forma de organizar, controlar y disponer el conocimiento existente sobre algún área, para que pueda ser utilizado.

B. Ontologías

Las ontologías son una herramienta flexible en la representación del conocimiento. Hendler (2001) las define como “un conjunto de términos de conocimiento, que incluye un vocabulario, relaciones y un conjunto de reglas lógicas y de inferencia sobre un dominio en particular”. Las ciencias de la computación, por su parte, las definen como la especificación explícita de un concepto (Gruber, 1995, p.2).

Las ontologías se componen de elementos, como las clases o conceptos del dominio, propiedades que se representan mediante relaciones o atributos; también constan de individuos o instancias de las mismas clases y axiomas, que corresponden a restricciones determinadas para uno o más elementos de la ontología.

Para la construcción de las ontologías es importante tener una buena organización de las clases en forma jerárquica, con el objetivo de que se encuentre entre las clases la relación clase-subclase y, así mismo, sea posible establecer que todos los individuos de una clase posean las mismas propiedades de su superclase. Es así como una base de conocimiento se compone, básicamente, de una ontología específica a un grupo de individuos (Noy & McGuinness, 2000).

La definición de Márquez (2007) es la más pertinente para este documento es la que se refiere a las ontologías

como la especificación compartida de un dominio de conocimiento que se puede comunicar entre personas y sistemas heterogéneos.

La implementación de una ontología se realiza mediante un lenguaje específico de representación, el cual debe ser elegido teniendo en cuenta las características necesarias para obtener una representación detallada y exacta. Mediante estos lenguajes es posible obtener representaciones: informales, expresadas en lenguaje natural; semi-formales, que permiten obtener representaciones en lenguaje máquina; y rigurosamente formales, con términos, reglas y relaciones semánticas definidas de manera explícita, proporcionando un alto grado de procesamiento computacional.

Entre los lenguajes computacionales útiles para la representación de ontologías se destaca el OWL [*Web Ontology Language*], desarrollado por el W3C Web Ontology Working Group, creado como una extensión del *Resource Description Framework* [RDF]. Es utilizado en el desarrollo de ontologías, para el uso compartido por la web (Ramírez, Alonso, Hernández, Arias, & La Rosa, 2010) y para especificar características como instancias y sus interrelaciones.

C. Lógicas descriptivas

Son un conjunto de representaciones semi-formales, constituidas por formalismos de representación de dominios basados en la lógica para la representación del conocimiento (Ruckhaus, 2005); permiten representar sus conceptos más relevantes junto con sus propiedades y relaciones.

Las lógicas descriptivas fortalecieron las debilidades semánticas presentes en otras formas de representación de conceptos, como los frames y las redes semánticas, en el área de la extracción de la información, como un uso importante de la representación del conocimiento (Tayal, Raghuwansh, & Latesh, 2013). A nivel computacional pueden ser trabajadas a través de razonadores, motores y reglas de inferencia (Holmes & Stocking, 2009). Las lógicas descriptivas pueden ser aplicadas a nivel abstracto, evaluando las relaciones conceptuales y, a nivel concreto, procesando las instancias disponibles, dando lugar a: T-box, para la descripción de conceptos jerárquicos o propiedades generales de los conceptos; y A-box, para establecer la relación de cada una de las instancias de los conceptos con su ubicación en la jerarquía (Flores, 2011).

La lógica de predicados hace parte de las lógicas descriptivas, lo cual facilita la representación de afirmaciones compuestas a través del uso de variables para representar objetos, utilizada también en campos de la inteligencia artificial (Ali & Khan, 2010). Cada una de las afirmaciones está representada por predicados que tienen un valor verdadero o falso (Parnas, 1993). Esta lógica tiene un alto grado de expresividad, por lo que permite representar el dominio de los trabajos de grado de las universidades, en su modalidad de monografía, expresando las condiciones procedimentales para una correcta realización de cada una de estas, facilitando la obtención de deducciones con informa-

which correspond to particular constraints for one or more of the ontology's elements.

For the building of ontologies, it is important to have a good hierarchical organization of the classes with the objective that the class-subclass relationship is within classes. Likewise it is necessary to have the possibility of establishing that all instances of a class have the same properties as their superclass. This is because a knowledge base is composed, basically, of a specific ontology for a group of instances (Noy & McGuinness, 2000).

Márquez's definition (2007) is the most appropriate for this document, as it refers to ontologies as the shared specification of a knowledge domain that can be communicated between people and heterogeneous systems.

Implementation of the ontology is performed through a specific language of representation, which must be chosen considering the features necessary to obtain an exact and detailed representation. Using these languages, it is possible to obtain informal representations, expressed by natural languages; as well as semi-formal languages that allow representations to be obtained in machine language; and strictly formal languages with terms, rules, and semantic relationships that can be defined in an explicit way, providing a high level of computational processing.

Among the useful computational languages for the representation of ontologies is highlighted OWL [*Web Ontology Language*], which was developed by the W3C Web Ontology Working Group and created as an extension of the *Resource Description Framework* [RDF]. This is utilized in the development of ontologies, for shared use by the web (Ramírez, Alonso, Hernández, Arias, & La Rosa, 2010) and to specify features as instances and their interrelationships.

C. Description logics

Description logics are a set of semi-formal representations, formed by domain representation formalisms based on the logic for the representation of knowledge (Ruckhaus, 2005); they allow us to represent their most relevant concepts, their properties and relationships.

Description logics strengthen the semantic weaknesses of other representations of concepts such as frames and semantic networks, in the area of the extraction of information with important use of knowledge representation (Tayal, Raghuwansh, & Latesh, 2013). This can be worked up to the computational level using reasoners, engines, and inference rules (Holmes & Stocking, 2009).

Applying description logics to the abstract level is possible, first, by evaluating the conceptual relationship, as well as to the concrete level by processing the available instances. This results in: T-box, for the description of hierarchical concepts or general properties of the concepts; and A-box, to establish the relationship for each of the instances of the concepts with its location in the hierarchy (Flores, 2011).

The predicate logic is part of the description logics, and facilitates the representation of composed affirmations using variables to represent objects, and is also used in the fields of artificial intelligence (Ali & Khan, 2010). Each affirmation is represented by predicates that are false or true values (Parnas, 1993). This logic has a high level of expressiveness, and therefore allows us to represent the monographs domain, expressing the conditions of the procedure for a correct relationship of each of these, and making it possible to obtain deductions using the information collected in a repository. In the above definition, information is understood as a collection of metadata, categorized as semantic or instructive information (Floridi, 2004).

The predicate logic of zero order is limited to the representation of affirmations or propositions that have a value set to true. However, it lacks structures such as quantifiers and variables that represent entities, and it is therefore impossible to represent statements such as “all men are immortals” or “Aristóteles is a man”, which can be represented through predicate logic of the first order. This concept is detailed below.

D. Predicate logic of the first order

Predicate logic of the first order is a formal language that uses elements such as constants, variables, functions and quantifiers to represent facts, which form the alphabet of this language. In this logic linguistically declarative statements are used that allow us to represent affirmations and facts. These can take true or false values using the objects' instances and their relationships, which constitutes a formal system of logic programming (Shi, Liu, & Chen, 2010).

The quantifiers used by the predicate logic of the first order are math symbols that allow us to establish the amount in a proposition, specifying the number of elements in a determined set that have a certain property. To specify that all the elements of a set meet a condition, this logic uses a quantifier of a universal type.

ción reunida en un repositorio. En la anterior definición, se entiende información como una colección de metadatos categorizada como información semántica o instructiva (Floridi, 2004).

La lógica de predicados de orden cero está limitada a la representación de afirmaciones o proposiciones que tienen un valor de verdad definido. Sin embargo, carece de estructuras como cuantificadores o variables que representen entidades, por lo que es imposible representar afirmaciones como “todos los hombres son mortales” o “Aristóteles es un hombre”, las cuales sí pueden ser representadas mediante la lógica de predicados de primer orden. Los conceptos inherentes a esta lógica se presentan en la siguiente sección.

D. Lógica de predicados de primer orden

La lógica de predicados de primer orden es un lenguaje formal que hace uso de elementos tales como: constantes, variables, funciones y cuantificadores, para representar hechos, los cuales forman el alfabeto de este lenguaje. A través de esta lógica se hace uso de enunciados, lingüísticamente declarativos, que permiten representar afirmaciones y hechos que pueden tomar un valor verdadero o falso, mediante las instancias u objetos y las relaciones entre ellos, haciendo de él un sistema formal de la programación lógica (Shi, Liu, & Chen, 2010).

Los cuantificadores que utiliza la lógica de predicados de primer orden son símbolos matemáticos que permiten establecer la cantidad en una proposición, especificando cuántos elementos de un determinado conjunto poseen una propiedad. Para especificar que todos los elementos de un conjunto cumplen una condición, esta lógica hace uso de un cuantificador de tipo universal. La especificación de que existe, como mínimo, un elemento de algún conjunto que cumple la condición, se hace a través del cuantificador existencial; finalmente, el existencial único establece que existe un único elemento de un conjunto que presenta alguna propiedad o condición. La obtención de formas lógicas es considerada como un paso intermedio entre el análisis sintáctico de expresiones y la formación de estructuras semánticas más profundas (Moldovan & Rus, 2001).

Esta lógica permite realizar deducciones mediante razonamientos de tipo $\{P_1, P_2, P_3, \dots\} \rightarrow q$, teniendo validez cuando la conclusión obtenida es una consecuencia de tipo lógico obtenida de las premisas evaluadas, premisas que a su vez están formadas por símbolos, denominados conectivos, dentro de los cuales se encuentra la negación, la cual permite indicar que un predicado es falso; la conjunción, que permite unir un predicado con otro; la disyunción, que une dos o más predicados y requiere que mínimo uno de estos sea verdadero para tener validez; la afirmación y la implicación, que establecen que si uno o varios predicados son válidos, ello implica la validez de un predicado final; y el conectivo bicondicional, que permite establecer la validez mutua entre dos o más predicados, simultáneamente, verdaderos o falsos. Todos estos conectivos permiten repre-

sentar afirmaciones o formulas a un buen detalle a través de los cuantificadores nombrados anteriormente (Ono, Kawano, Fukazawa, & Kadokura, 1992); cada uno de los predicados puede tener uno o más parámetros para representar información, como puede ser la premisa $A(x)$ o también $B(X_1, X_2, \dots)$.

Como parte atómica de esta lógica se encuentran las clases, las cuales corresponden a conceptos o entidades que se utilizan en la ontología, y son representadas mediante instancias o elementos individuales de una clase en particular. Los predicados que reúnen dos o más elementos son conocidos como relaciones; finalmente, se encuentran los axiomas formales, que son modelamientos de afirmaciones que se consideran siempre verdaderas, que son utilizadas para validar ciertos predicados y así realizar las deducciones deseadas.

III. Metodología

La construcción de la base de conocimiento implicó el análisis y diseño incremental, desde una perspectiva ontológica, de la herramienta de modelamiento, el lenguaje utilizado para la representación computacional, expresando las clases –y las relaciones entre clases– de la ontología. Finalmente, se expresaron las reglas de la base de conocimiento mediante lógica de predicados en notación algebraica para luego obtener las inferencias, como se describe en las siguientes secciones.

A. Planteamiento de la ontología

Para el desarrollo de la ontología se utilizó la versión de software Protege 5.0, con él se construyó la ontología en OWL; para la verificación de la consistencia de la ontología se hizo uso del razonador Racer pro, que soporta la versión FULL de OWL e implementa la lógica descriptiva SHIQ (Dentler, Cornet, Teije & Keiser, 2011).

Para determinar los conceptos fundamentales a utilizar en la ontología, se usaron documentos y acuerdos generados por universidades que describen los detalles sobre el proceso de realización de la monografía como modalidad de grado. Esta ontología será utilizada en la representación de la base de conocimiento, donde se explicarán algunas de las relaciones mediante la formulación de reglas en lógica de predicados, verificando que realmente se cumplan las relaciones propuestas en la ontología.

La ontología utilizada para el desarrollo de este trabajo se presenta mediante un modelo en forma de grafo. Los nodos *Jurado*, *Director* y *Anteproyecto*, representan clases, las cuales pueden ser referenciadas por instancias, mientras que los arcos *evaluadoPor*, *tieneTema* y *dirigidoPor*, representan relaciones que hacen explícitas las asociaciones entre conceptos, como se presenta en la FIGURA 1.

B. Representación de la ontología en OWL

Las relaciones entre conceptos pueden ser representadas mediante un lenguaje que puede ser ejecutable por máquina, como OWL. La representación ontológica en OWL es

The existential quantifier is one which allows us to have a specification of the existence of a minimum of one element from some set satisfying the criteria; finally, the unique existential quantifier establishes that there exists a unique element from a set that represents some property or condition. Obtaining logic forms is considered as an intermediate step between the syntactic analysis of expressions and the formation of deeper semantic structures (Moldovan & Rus, 2001).

This logic allows us to make deductions via reasoning of type $\{P1, P2, P3...\} \rightarrow q$, which is valid if the conclusion obtained from the evaluated premises is of the logic type. These premises are formed by symbols called connectives, including the negation that allows a false value to be set for a predicate; the conjunction, which joins one predicate with another; the disjunction that joins two or more predicates and requires at least one true value for its validity; affirmation and implication, which establish that if one or more predicates are valid this implies the validity of a final predicate; and the bi-conditional connective that allows us to verify the mutual validity between two or more predicates simultaneously, with true or false values. All the connectives allow us to represent statements or formulas in clear detail using the quantifiers above (Ono, Kawano, Fukazawa, & Kadokura, 1992); each of the predicates can have one or more parameters to represent information, for example, the premise $A(x)$ or $B(X1, X2, \dots)$.

The classes, which are the atomic part of this logic, are concepts or entities used in the ontology and are represented by instances or individual elements from a particular class. The predicates that join two or more elements are known as relationships; finally, the formal axioms are the models of affirmations that always take true values, used to validate some predicates and then make the desired deductions.

II. Methodology

The development of the knowledge base represents an incremental design and analysis from an ontological perspective of the modeled tool, with the language used for the computational representation expressing the classes – and their relationships - of the ontology. Finally, the rules of the knowledge base were expressed using the predicate logic in algebraic notation to later obtain the inferences described in the next sections.

A. Approach of the ontology

In the development of the ontology in OWL, version 5.0 of Protege software and Racer Pro reasoner were used to verify the ontology's consistence. This reasoner supports the full version of OWL using SHIQ (Dentler, Cornet, Teije & Keiser, 2011) for the implementation of the descriptive logic.

The documents and agreements generated by universities that describe the details of the monograph development process in degree project modality were used to determine the fundamental concepts to use in the ontology. The representation of the knowledge base will use this ontology, where some relationships are defined via the formulation of rules in predicate logic, verifying that the proposed relationships in the ontology are really followed.

The ontology used in the development of this paper is presented employing a model in the form of a graph. The nodes *Jurado*, *Director* and *Anteproyecto* represent classes, which can be referenced by instances, while the arcs *evaluadoPor*, *tieneTema* and *dirigidoPor* represent relationships that convert the associations between concepts to explicit form, as shown in **FIGURE 1**.

B. Ontology representation in OWL

The relationships between concepts are represented via a language that can be executable by machine, such as OWL. The ontological representation in OWL is as follows:

- Classes: each one of the entities represented in the ontology, some of which are presented as follows:

Class(: TrabajoDeGrado)

Class(: Tema)

- Individuals: own instances of ontology; some examples are presented as follows:

Individual(:Monografia :1)

Individual(:Tema :Pruebas)

- Properties: The relationships existing between the entities; these are structured by the *ObjectProperty*, where their domain and range are specified; some relationships are presented as follows:

ObjectProperty(: hechoPor
domain (: TrabajoDeGrado)
range (: Estudiante))

ObjectProperty(: validadopor
domain (: TrabajoDeGrado)
range (: Revisor))

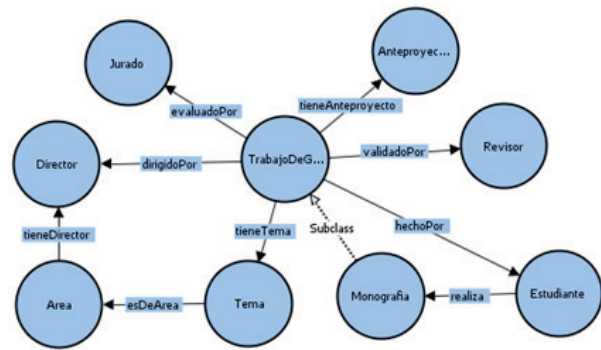


Figure 1. Ontology used / Ontología utilizada

la siguiente:

- Clases: cada una de las entidades representadas en la ontología; algunas de ellas se representan de la siguiente manera:

Class(: TrabajoDeGrado)

Class(: Tema)

- Individuos: instancias propias de la ontología; algunos ejemplos de ellos se representan de la siguiente manera:

Individual(:Monografia :1)

Individual(:Tema :Pruebas)

- Propiedades: relaciones existentes entre las entidades; su estructura se compone de elementos como *ObjectProperty*, donde se especifica su dominio (*domain*) y su rango (*range*); algunas de estas relaciones se representan de la siguiente manera:

ObjectProperty(: hechoPor
domain (: TrabajoDeGrado)
range (: Estudiante))

ObjectProperty(: validadopor
domain (: TrabajoDeGrado)
range (: Revisor))

En este caso, la primera propiedad tiene como nombre *hechoPor*, y establece la relación entre clases de tipo *TrabajoDeGrado* y clases de tipo *Estudiante*, lo que significa que un trabajo de grado es hecho por un estudiante; igual sucede con la segunda propiedad presentada.

- Subclases: usan la estructura *subClassOf*; el caso usado es el siguiente:

subClassOf(:TrabajoDeGrado :Monografia)

En este caso, se dice que la clase *Monografia* es una subclase de la clase *TrabajoDeGrado*.

- Relación 'uno de': enumeradores considerados como un tipo de dato antes de ser considerados como clases; ejemplos que podrían existir en la base de conocimiento serían los siguientes:

Class(: Tema

OneOf(: Pruebas:IngenieríaDeSoftware))

Class(: Director

OneOf(: JulianTorres:CamiloSantos))

En este caso, se especifica que la clase *Tema* puede tomar el nombre *Pruebas* o *IngenieríaDeSoftware* como identificador en su nombre.

- Cardinalidad: mediante OWL es posible establecer las cardinalidades entre las relaciones, permitiendo dar el número mínimo (*minCardinality*), el máximo (*maxCardinality*) o la cantidad exacta (*cardinality*), de las instancias utilizadas por cada relación. La estructura base es de la forma (cantidad, relación, entidad rango). Algunas de estas se representan de la siguiente manera:

mincardinality(1 :hechoPor:TrabajoDeGrado)

maxcardinality(1 :validadoPor:TrabajoDeGrado)

La primera instrucción establece que la clase *TrabajoDeGrado*, mediante la relación *hechoPor*, debe estar relacionada con mínimo un elemento de su rango que corresponde a *Estudiante*. La segunda instrucción establece que la clase *TrabajoDeGrado*, mediante la relación *validadoPor*, puede estar relacionada con máximo un único elemento de su rango, que corresponde a *Revisor*.

Estas especificaciones fueron obtenidas del modelo presentado en la sección 3.1.

C. Planteamiento de la base de conocimiento

La base de conocimiento propuesta está formada por reglas que se basan en las relaciones de la ontología. A continuación se propone una representación en lógica de predicados referente a la modalidad monografía, para la cual se consultó a algunas universidades, obteniendo las siguientes reglas. Algunas de ellas se explican para facilitar su interpretación.

- Toda monografía es realizada por al menos un estudiante:

$\forall x(Monografía(x)) \rightarrow \exists y / (Estudiante(y) \wedge EsTrabajoDeGradoDe(x,y))$

Esta sentencia establece que por cada elemento de la clase *Monografía*, debe existir un elemento de tipo *Estudiante*, donde el estudiante esté relacionado con el elemento de tipo *Monografía* mediante la relación *EsTrabajoDeGradoDe*. Esta relación se expresa de manera explícita dado que se requiere la verificación real de que cada monografía tiene mínimo un estudiante que la está realizando o la realizó.

La verificación del predicado *Estudiante(y)* hace explícita la relación entre estudiante y monografía. Es posible escribir la regla omitiendo esta parte, siempre y cuando los hechos definidos en la base de conocimiento cumplan los patrones de cada relación, como se especifica en la Tabla 1, ubicada al final de la sección. De esta manera la relación

In this case the first property has the name *hechoPor*, and establishes the relationship between classes of type *TrabajoDeGrado* and classes of type *Estudiante*. This means that a degree project (instance of *TrabajoDeGrado* class) is performed (*hechoPor* range) by a student (instance of *Estudiante* class), and likewise for the second property presented.

- Subclasses: these use the *subClassOf* structure, as shown below:

subClassOf(:TrabajoDeGrado :Monografía)

In this case the *Monografía* Class is a subclass of the *TrabajoDeGrado* class.

- Relation ‘OneOf’: these are enumerators considered as a data type instead of as classes, for example:

Class(: Tema

OneOf(: Pruebas:IngenieríaDeSoftware))

Class(: Director

OneOf(: JulianTorres:CamiloSantos))

In this case it is specified that the *Tema* class can be called *Pruebas* or *IngenieríaDeSoftware*, for the identifier of the name.

- Cardinality: In OWL it is possible to establish cardinality between two relationships, specifying the minimum number (*minCardinality*), the maximum number (*maxCardinality*) or the precise amount (*cardinality*) of the instances used by each relationship. The structure (amount, relationship, range entity) is used to represent the base structure, for example:

mincardinality(1 :hechoPor:TrabajoDeGrado)

maxcardinality(1 :validadoPor:TrabajoDeGrado)

The first instruction establishes that the *TrabajoDeGrado* class must be related at least once to an element of its range that corresponds to *Estudiante*, through the *hechoPor* relationship. The second instruction establishes that the *TrabajoDeGrado* class must at most be related to one element of its range that corresponds to *Revisor*.

These specifications were obtained from the model presented in section 3.1.

C. The knowledge base approach

The proposed knowledge base is formed by rules based on the ontology's relationship. Below is proposed a representation in predicate logic referring to the monograph mode. For this, some universities were consulted

to obtain the following results, some of which are explained to facilitate its interpretation.

- All monographs are performed by at least one student:

$$\forall x(\text{Monografia}(x)) \rightarrow \exists y/ (\text{Estudiante}(y) \wedge \text{EsTrabajoDeGradoDe}(x,y))$$

This sentence establishes the existence of one element of *Estudiante* type for each element of the *Monografia* class where a student (instance of *Estudiante*) is related with the element of *Monografia* type by the *EsTrabajoDeGradoDe* relationship. Given that there is the need for a real verification that least at one student performs or performed the monograph, this relationship is expressed in an explicit way.

The relationship between student and monograph becomes explicit by the verification of the *Estudiante*(*y*) predicate. The rule can be written without this part, ensuring that the defined facts in the knowledge base follow the pattern of each relationship as shown in Table 1 at the end of the section. In this way the relationship *EsTrabajoDeGradoDe* requires that the first argument must be of *Monografia* type and the second of *Estudiante* type. Although this type of specification improves the rule description, some sentences are omitted because they do not contribute to the final result; furthermore they would require more extension and processing

- Every monograph is considered a degree work:

$$\forall x(\text{Monografia}(x)) \rightarrow \exists y/ (\text{TrabajoDeGrado}(y) \wedge \text{Igual}(x,y))$$

- Every monograph has a specific theme:

$$\forall x(\text{Monografia}(x)) \rightarrow \exists y/ (\text{Tema}(y) \wedge \text{EsTemaDe}(y,x))$$

- Every monograph has a proposed draft (Universidad del Rosario, 2011):

$$\forall x(\text{Monografia}(x)) \rightarrow \exists y/ (\text{Anteproyecto}(y) \wedge \text{EsAnteproyectoDe}(y,x))$$

- The proposed draft must be sent to the director of the research group (Universidad del Rosario, 2011):

$$\forall x,y(\text{EsAnteproyectoDe}(y,x)) \rightarrow \exists d/ (\text{DirectorDeGrupoInv}(d) \wedge \text{Notificación}(y,d))$$

As in the first rule, the left part is omitted - where it verifies that the type of element *x* must be *Monografia* and the element *y* of *Anteproyecto* type - because the knowledge represented as facts in the base must follow the pattern of the input data in the predicate *EsAnteproyectoDe*. However, the verification that the *d* element's type is equal to *DirectorDeGrupoInv* is explicit in order to ensure that the class of the receptor element is equal.

EsTrabajoDeGradoDe requiere que el primer argumento sea de tipo *Monografia* y el segundo de tipo *Estudiante*. Aunque este tipo de especificaciones mejoran la descripción de la regla, se omitieron en algunas sentencias, dado que no aportan al resultado final y requieren mayor extensión y exigencia de procesamiento.

- Toda monografía es considerada trabajo de grado:

$$\forall x(\text{Monografia}(x)) \rightarrow \exists y/ (\text{TrabajoDeGrado}(y) \wedge \text{Igual}(x,y))$$

- Toda monografía tiene un tema específico:

$$\forall x(\text{Monografia}(x)) \rightarrow \exists y/ (\text{Tema}(y) \wedge \text{EsTemaDe}(y,x))$$

- Toda monografía tiene anteproyecto (Universidad del Rosario, 2011):

$$\forall x(\text{Monografia}(x)) \rightarrow \exists y/ (\text{Anteproyecto}(y) \wedge \text{EsAnteproyectoDe}(y,x))$$

- El anteproyecto debe ser enviado al director del grupo de investigación (Universidad del Rosario, 2011):

$$\forall x,y(\text{EsAnteproyectoDe}(y,x)) \rightarrow \exists d/ (\text{DirectorDeGrupoInv}(d) \wedge \text{Notificación}(y,d))$$

Al igual que en la primera regla, se omite en la parte izquierda verificar que el elemento *x* sea de tipo *Monografia* y el elemento *y* del tipo *Anteproyecto*, pues el conocimiento representado como hechos en la base de conocimiento debe seguir con el patrón de datos ingresado con el predicado *EsAnteproyectoDe*. Sin embargo, se dejó explícita la verificación de que el elemento *d* debe ser de tipo *Director* de un grupo de investigación, con el fin de garantizar que el elemento receptor de la notificación sea de esta misma clase.

- El resultado puede ser aprobado, no aprobado o modificable (Universidad del Rosario, 2011):

$$\forall x,y,d(\text{Anteproyecto}(x) \wedge \text{Notificación}(y,d)) \rightarrow \exists x/ (\text{Aprobación}(x, \text{'viable'}) \vee \text{Aprobación}(x, \text{'modificable'}) \vee \text{Aprobación}(x, \text{'noViable'}))$$

- Si es aprobado, se debe terminar en un año máximo (Universidad del Rosario, 2011):

$$\forall x(\text{Anteproyecto}(x) \wedge \text{Aprobación}(x, \text{'viable'}) \wedge \text{FechaAprobación}(x, f)) \rightarrow \exists m(\text{EsAnteproyectoDe}(x,m) \wedge \text{Finalizada}(m, f_2) \wedge f_2 \leq f + 365)$$

Esta regla establece que, por cada elemento de la clase *Anteproyecto* que haya sido aprobado o considerado viable en una fecha *f*, debe existir una segunda fecha en la que haya sido finalizada la monografía *x* a la que pertenecía el anteproyecto en cuestión. Además, se evalúa que esta segunda fecha *f₂* sea posterior a la fecha inicial y esté en un período no mayor a 365 días de ella.

- Una vez realizada por el estudiante, se asignan 2 jurados del proyecto (Universidad del Rosario, 2011):

$$\forall x(\text{Monografia}(x) \wedge \text{Finalizada}(x, f)) \rightarrow \exists j,k (\text{EsJuradoDe}(j,x) \wedge \text{EsJuradoDe}(k,x) \wedge \neg \text{Igual}(j,k))$$

En este caso se evalúa que, por cada monografía *x* finalizada en una fecha *f*, existan dos jurados *j*, *k* diferentes, verificado mediante la expresión $\neg \text{Igual}(j,k)$. A la vez estos

dos elementos j , k deben ser jurados de la monografía x . Esta última condición se evalúa mediante el predicado *EsJuradoDe*, el cual contiene, como argumentos, al jurado y a la monografía correspondiente, en ese mismo orden.

- El concepto debe ser modificable o sustentable (Universidad del Rosario, 2011):

$$\forall m, x, y \ (EsJuradoDe(x, m) \wedge EsJuradoDe(y, m) \wedge \neg Igual(x, y)) \rightarrow \exists x / (Cierre(x, 'modificable') \vee Cierre(x, 'sustentable'))$$

- Si es sustentable, debe asignar una fecha de sustentación (Universidad del Rosario, 2011):

$$\forall m (Finalizada(m, f) \wedge Cierre(x, 'sustentable') \rightarrow \exists f_2 (Sustentación(m, f_2) \wedge f_2 \geq f)$$

- El director debe asistir a la sustentación (Universidad del Rosario, 2011).

$$\forall m (Sustentación(m, f) \rightarrow \exists d (Director(d) \wedge Asistencia(d, m))$$

- Los jurados deben asistir a la sustentación (Universidad del Rosario, 2011):

$$\forall m (Sustentación(m, f) \rightarrow \exists j, k (Jurado(j) \wedge Jurado(k) \wedge \neg Igual(j, k) \wedge Asistencia(j, m) \wedge Asistencia(k, m))$$

- El coordinador de trabajos de grado o un miembro del comité de trabajos de grado debe asistir a la sustentación (Universidad del Rosario, 2011):

$$\forall m (Sustentación(m, f) \rightarrow \exists d ((Coordinador(d) \vee MiembroComité(d)) \wedge Asistencia(d, m))$$

Se evalúa que, por cada sustentación de una monografía m asignada en una fecha f , quede registrado que un elemento d de tipo Coordinador o MiembroComité haya asistido a la sustentación de la monografía m , cuando esta fue asignada.

La Tabla 1 presenta los predicados utilizados y la explicación de los argumentos que los componen.

IV. Resultados

Con el conocimiento planteado a través de las reglas y los hechos mostrados como ejemplo, es posible realizar inferencias y extraer información implícita en la base de conocimiento, que sea de utilidad para el usuario. A continuación se presentan las inferencias obtenidas, expresando las premisas utilizadas de la ontología y algunas de las planteadas junto con la conclusión obtenida en lenguaje natural y en notación algebraica.

A. Relación monografía-revisor

P_1 : Toda monografía es considerada trabajo de grado:

$$\forall x (Monografía(x)) \rightarrow \exists y / (TrabajoDeGrado(y) \wedge Igual(x, y))$$

P_2 : Todo trabajo de grado tiene un revisor:

$$\forall x (TrabajoDeGrado(x)) \rightarrow \exists y / (EsRevisorDe(y, x))$$

C: Toda monografía tiene un revisor:

$$\forall x (Monografía(x)) \rightarrow \exists y / (EsRevisorDe(y, x))$$

Donde se deduce que, por cada elemento de la clase Mo-

- The result may be approved, not approved or modifiable (Universidad del Rosario, 2011):

$$\forall x, y, d (Anteproyecto(x) \wedge Notificación(y, d) \rightarrow \exists x / (Aprobación(x, 'viable') \vee Aprobación(x, 'modificable') \vee Aprobación(x, 'noViable'))$$

- If it is modifiable, it must be completed in one year at most (Universidad del Rosario, 2011):

$$\forall x (Anteproyecto(x) \wedge Aprobación(x, 'viable') \wedge FechaAprobación(x, f) \rightarrow \exists m (EsAnteproyectoDe(x, m) \wedge Finalizada(m, f_2) \wedge f_2 \leq f + 365))$$

This rule establishes that for each element of the *Anteproyecto* class, which has been approved or considered viable on a date f , there must also exist a second date for the monograph x to which the proposed draft belongs. Furthermore, the second date f_2 is evaluated to verify that it is later than the initial date, and does not exceed it by more than 365 days.

- Two panels of judges (Jurado) are assigned to the project after completion of the monograph. (Universidad del Rosario, 2011):

$$\forall x (Monografía(x) \wedge Finalizada(x, f) \rightarrow \exists j, k (EsJuradoDe(j, x) \wedge EsJuradoDe(k, x) \wedge \neg Igual(j, k))$$

For each completed monograph at a date f , the existence of two different panels j , k is verified using the expression $\neg Igual(j, k)$. Additionally, these two elements (j , k) must be the panels for the monograph. This last condition is evaluated using the *EsJuradoDe* predict, where the arguments are the panel and the corresponding monograph in that order.

- The concept can be modifiable or sustainable (Universidad del Rosario, 2011):

$$\forall m, x, y (EsJuradoDe(x, m) \wedge EsJuradoDe(y, m) \wedge \neg Igual(x, y)) \rightarrow \exists x / (Cierre(x, 'modificable') \vee Cierre(x, 'sustentable'))$$

- If it is sustainable, a sustainment date must be assigned (Universidad del Rosario, 2011):

$$\forall m (Finalizada(m, f) \wedge Cierre(x, 'sustentable') \rightarrow \exists f_2 (Sustentación(m, f_2) \wedge f_2 \geq f)$$

- The director must attend to the sustainment (Universidad del Rosario, 2011).

$$\forall m (Sustentación(m, f) \rightarrow \exists d (Director(d) \wedge Asistencia(d, m))$$

- Panels must attend to the sustainment (Universidad del Rosario, 2011):

Table 1. Parameters of the recovery function in the SIR model / Parámetros de la función de recuperación del modelo SIR

Predicate name	First argument / Primer argumento	Second argument / Segundo argumento
EsTrabajoDeGradoDe	Monograph as subclass of TrabajoDeGrado / Monografía como subclase de TrabajoDeGrado	Student who performs the monograph. / Estudiante que realiza la monografía
EsTemaDe	Theme / Tema	Monograph which addresses the assigned theme. / Monografía que trata el tema dado
EsCarreraDe	Career / Carrera	Student who belongs to the referenced career. / Estudiante que pertenece a la carrera dada.
EsAnteproyectoDe	Draft of project / Documento anteproyecto.	Monograph corresponding to the draft indicated. / Monografía correspondiente al anteproyecto indicado
Notificación	Document sent in the notification / Documento enviado en la notificación.	Person who receives the notification. / Quién recibe la notificación
Aprobación	Draft of monograph / Anteproyecto de monografía.	Result of initial evaluation. / Resultado de la evaluación inicial
FechaAprobación	Draft approved. / Anteproyecto aprobado.	Date when draft was approved. / Fecha en que se aprobó el anteproyecto.
EsJuradoDe	Teacher who is the assigned panel for the final presentation of the project. / Docente que actuara como jurado de la sustentación.	Monograph assigned to the panel indicated. / Monografía que tiene asignado al jurado anterior.
Cierre	Draft of monograph. / Anteproyecto de monografía.	Result of evaluation of the final degree work presentation. / Resultado de la evaluación sobre la sustentación.
Finalizada	Monograph degree work. / Trabajo de grado monografía.	Date when the project was finalized. / Fecha en la cual fue oficialmente terminada.
Sustentación	Monograph degree work. / Trabajo de grado monografía.	Date when the presentation was done. / Fecha de sustentación de la monografía.
Asistencia	Person who attends to a final presentation of degree work. / Persona que asiste a una sustentación.	Monograph corresponding to the sustainment. / Monografía correspondiente a la sustentación.

$$\forall m(\text{Sustentación}(m, f)) \rightarrow \exists j, k (\text{Jurado}(j) \wedge \text{Jurado}(k) \wedge \neg \text{Igual}(j, k) \wedge \text{Asistencia}(j, m) \wedge \text{Asistencia}(k, m))$$

- The degree work coordinator or the member of the degree work committee must attend to the sustainment (Universidad del Rosario, 2011):

$$\forall m(\text{Sustentación}(m, f)) \rightarrow \exists d ((\text{Coordinador}(d) \vee \text{MiembroComité}(d)) \wedge \text{Asistencia}(d, m))$$

For each monograph assigned to f date, an element d of either *Coordinador* or *MiembroComité* type is registered to attend the presentation of monograph m , when that is assigned.

TABLE 1 presents the predicates used and explains the arguments of their composition.

IV. Results

With the proposed knowledge through rules, and the facts reported as an example, it is possible to make inferences and extract implicit information useful to the user from the knowledge base. The resulting inferences are presented below, expressing the assumptions used in the ontology and submitted beside the conclusion in the natural language and the algebraic notation.

nografía existente, debe existir también un revisor que la valide.

B. Relación monografía-jurado

P_1 : Toda monografía es considerada trabajo de grado:

$$\forall x(\text{Monografía}(x)) \rightarrow \exists y / (\text{TrabajoDeGrado}(y) \wedge \text{Igual}(x, y))$$

P_2 : Todo trabajo de grado tiene un jurado:

$$\forall x(\text{TrabajoDeGrado}(x)) \rightarrow \exists y / (\text{EsJuradoDe}(y, x))$$

C: Toda monografía tiene un jurado:

$$\forall x(\text{Monografía}(x)) \rightarrow \exists y / (\text{EsJuradoDe}(y, x))$$

En este caso se verifica que, por cada elemento de la clase *Monografía*, exista un elemento de la clase *Jurado* que cumpla la función de evaluador de esta.

Estas dos inferencias permitirían mejorar los procesos de registro al evaluar si los datos o el conocimiento se han ingresado de manera correcta. De manera similar a las verificaciones anteriores si se encuentra en la base de conocimiento el hecho *EsRevisorDe*(y, x) o *EsJuradoDe*(y, x), esta deduciría que deben existir los hechos *TrabajoDeGrado*(y) y *Monografía*(x), agregándolos a la lista de hechos existentes en ese momento. Otra alternativa consiste en presentar alertas sobre inconsistencias cuando alguno de estos hechos no corresponda, advirtiendo que es necesario realizar una revisión por parte de expertos para evaluar si se presenta-

ron errores o ambigüedades en el registro.

B. Categorización de las monografías existentes

A través de la clasificación de temas por cada área que exista es posible mejorar los procesos de consulta realizados a la base de conocimiento que contenga los hechos registrados acerca de las monografías, una vez se haya indicado el tema específico de cada una. La inferencia obtenida en este caso es la siguiente:

P₁: Toda monografía tiene un tema específico:

$$\forall x(\text{Monografía}(x)) \rightarrow \exists y / (\text{Tema}(y) \wedge \text{EsTemaDe}(y,x))$$

P₂: Todo tema pertenece a un área general:

$$\forall x(\text{Tema}(x)) \rightarrow \exists y / (\text{Área}(y) \wedge \text{PerteneceA}(x,y))$$

C: Toda monografía pertenece a un área general:

$$\forall x(\text{Monografía}(x)) \rightarrow \exists y / (\text{Área}(y) \wedge \text{PerteneceA}(x,y))$$

Esta inferencia permitiría a la base de conocimiento presentar una lista de temas específicos registrados inicialmente para ser consultados. Una vez se seleccione algún tema, la base de conocimientos obtendrá la inferencia al elegir, de manera dinámica, las monografías correspondientes a ese tema y a cada uno de los temas pertenecientes al área del tema inicial, con el fin de que la base de conocimiento esté clasificada mediante taxonomías y a la vez sea más fácil consultar documentos que sean realmente relevantes y aporten a la búsqueda realizada o al proyecto de investigación que se esté llevando a cabo.

V. Conclusiones

El uso de herramientas como las bases de conocimiento en universidades o incluso gestores de enciclopedias online es muy útil, tanto para sus creadores, como para los usuarios, encontrando errores rápidamente en el manejo de la información, facilitando la visibilidad de resultados relevantes y mejorando las deficiencias que presentan las bases de datos tradicionales, en las cuales los resultados de las consultas son de tipo sintáctico o textual, más no semántico. Adicionalmente, permite encontrar errores con mayor rapidez gracias a la aplicación de la lógica de predicados, dado que esta permite representar el conocimiento de una manera en que se puedan establecer reglas y realizar inferencias para obtener información no representada de manera explícita.

La aplicación de las bases de conocimiento en el dominio de trabajos de grado permite representar, a través de elementos de tipo lógico, ontológico y computacional, procesos de manejo de información referente a las monografías, así como obtener mejores resultados al momento de hacer búsquedas detalladas, mediante la presentación de resultados con alto grado de pertinencia para el usuario. Lo anterior es muy útil, tanto para los administradores de la base de conocimiento, como para los usuarios, al encontrar inconsistencias en el manejo de la información y mostrar resultados relevantes de búsqueda, resolviendo así problemas que presentan, generalmente, las bases de datos tradicionales, las cuales, como se dijo, hacen consultas de manera sintáctica o textual, no semántica.

A. Relationship monograph-reviewer

P₁: Any monograph is considered a degree work:

$$\forall x(\text{Monografía}(x)) \rightarrow \exists y / (\text{TrabajoDeGrado}(y) \wedge \text{Igual}(x,y))$$

P₂: Any degree work has a reviewer:

$$\forall x(\text{TrabajoDeGrado}(x)) \rightarrow \exists y / (\text{EsRevisorDe}(y,x))$$

C: Any monograph has a reviewer:

$$\forall x(\text{Monografía}(x)) \rightarrow \exists y / (\text{EsRevisorDe}(y,x))$$

It follows that for each element of the *Monografía* class there must also exist a reviewer who validates it.

B. Relationship monograph-panel

P₁: Any monograph is considered a degree work:

$$\forall x(\text{Monografía}(x)) \rightarrow \exists y / (\text{TrabajoDeGrado}(y) \wedge \text{Igual}(x,y))$$

P₂: Any degree work has a panel of judges:

$$\forall x(\text{TrabajoDeGrado}(x)) \rightarrow \exists y / (\text{EsJuradoDe}(y,x))$$

C: Any monograph has a panel of judges:

$$\forall x(\text{Monografía}(x)) \rightarrow \exists y / (\text{EsJuradoDe}(y,x))$$

In this case the existence is verified of one element from the *Jurado* Class (panel) for each element of the *Monografía* class, who fulfils the function of evaluator.

These two inferences allow us to improve the recording processes to evaluate data or the knowledge recorded in a correct way. Similarly to the above verifications, if in the knowledge base the fact *EsRevisorDe*(y, x) or *EsJuradoDe*(y, x) is found, it can be deduced that the facts *TrabajoDeGrado*(y), and *Monografía*(x) must exist, adding them to the list of existing facts at this time. Another alternative is based on presenting alerts about inconsistencies when some of the facts do not correspond, accompanied by a warning that a review by experts is needed to evaluate the existence of errors or ambiguities in the record.

C. Categorization of existing monographs

Using theme classification by each existing area, it is possible to improve the querying processes. These processes are performed on the knowledge base with the registered facts about the monographs, when the specific theme of each one is defined. The obtained inference is as below:

P₁: Any monograph has a specific theme:

$$\forall x(\text{Monografía}(x)) \rightarrow \exists y / (\text{Tema}(y) \wedge \text{EsTemaDe}(y,x))$$

P₂: Any item belongs to a general area:

$$\forall x(\text{Tema}(x)) \rightarrow \exists y / (\text{Área}(y) \wedge \text{PerteneceA}(x,y))$$

C: Any monograph belongs to a general area:

$$\forall x(Monografía(x)) \rightarrow \exists y / (Área(y) \wedge PertenecerA(x,y))$$


The knowledge base through inference will present a list of specific themes, which are recorded initially for consultation. When a theme is chosen, the knowledge base will receive the inference. This choice is dynamic and includes the monographs corresponding to this theme and each of the themes belonging to the area of the initial theme. Then the aim is to categorize the knowledge base using taxonomies, to query the relevant documents more easily, and finally to contribute to the search performed or the research project involved.

V. Conclusions


The use of tools such as knowledge bases at universities or even by administrators of on-line encyclopedias is very useful, for both its creators and users. It allows errors in the information management to be found quickly, facilitating the visibility of relevant results and improving the efficiency with regard to traditional databases, in which the results of queries are of a syntactic or textual type but not semantic. In addition, it allows errors due to the predicate logic to be found more quickly, which enables the knowledge representation with rules and performs inferences for obtaining implicit information.

The application of knowledge bases in the domain of degree works allows us to represent, using logic type, oncology and computational elements, the processes of information management referring to monographs. Also, the best results are retrieved at the moment of searching in detail, using the representation of results with a high level of relevance for the user. This is very useful for both knowledge base administrators and users, when inconsistencies in the information management are found and relevant results in searches can be shown. Thus, it is possible to solve the problems of traditional databases that relate to the fact that their queries are syntactic and textual instead of semantic.

VI. Recommendations and future work

This system should be a research project, improving the inference engines to find and repair inconsistency problems in the knowledge in a fast and dynamic way with the possibility of developing expert systems for the presented proposals. 

VI. Recomendaciones y trabajo futuro

Estos sistemas podrían ser un proyecto de investigación, mejorando los motores de inferencia para que encuentren y reparen problemas de inconsistencias en el conocimiento de manera rápida y dinámica, pues existe la posibilidad de desarrollar sistemas expertos para los propósitos presentados. 

References / Referencias

- Ali, A. & Khan, M. A. (2010, October). Knowledge representation of Urdu text using predicate logic. In *2010 6th International Conference on Emerging Technologies (ICET)*, (pp. 293-298). IEEE.
- Alonso, M. (1992). *Conocimiento y bases de datos una propuesta de integración inteligente* [thesis]. Universidad de Cantabria: Santander, España.
- Beekman, G. (1999). *Introducción a la computación*. Madrid, España: Pearson.
- Citycorp. (2015). *Semantic solutions for your enterprise*. Retrieved from: <http://www.cyc.com/enterprise-solutions/solutions/>
- Cybertesis [web site] (2015). Retrieved from: <http://cybertesis.unmsm.edu.pe/>
- Dentler, K., Cornet, R., Teije, A., & Keizer, N. (2011). Comparison of reasoners for large ontologies in the OWL 2 EL Profile. *Semantic Web*, 2(2), 71-87.
- Díez, D. & Zúñiga, A. (2011). Implementación de un modelo de gestión del conocimiento para empresas de servicios [thesis]. Cali, Colombia: Universidad ICESI.
- Flores, I. (2011). *Introducción al razonamiento sobre ontologías*. Retrieved from: <file:///C:/Users/Jos%C3%A9%20Ignacio/Downloads/nota-docencia-9.pdf>
- Floridi, L. (2004). Open problems in the philosophy of information. *Metaphilosophy*, 35(4), 554-582.
- Gangemi, A., Pisanelli, D., & Steve, G. (1998). Ontology integration: Experiences with medical terminologies. In *Formal ontology in information systems* [Vol. 46]. Amsterdam, The Netherlands: IOS.
- Gruber, T. (1995). Toward principles for the design of ontologies used for knowledge sharing. *International Journal of Human and Computer Studies. International journal of human-computer studies*, 43(5), 907-928.
- Hao, C. (2011). Research on knowledge model for ontology-based knowledge base. In *2011 International Conference on Business Computing and Global Informatization (BCGIN)*, (pp. 397-399). IEEE.
- Hendler, J. (2001). Agents and the semantic web. *IEEE Intelligent Systems*, 2, 30-37.
- Holmes, D. & Stocking, R. (2009). Augmenting agent knowledge bases with OWL ontologies. 2009 IEEE Aerospace Conference. doi:10.1109/AERO.2009.4839651
- Lahaba, N. & León, M. (2001). La gestión del conocimiento: una nueva perspectiva en la gerencia de las organizaciones. *Acimed*, 9(2), 121-126.
- Lenat, D., & Guha, R. V. (1990). Cyc: A midterm report. *AI magazine*, 11(3), 32.
- Márquez, S. (2007). *La web semántica* [thesis]. Universidad Politécnica de Madrid, España.
- Moldovan, D. I., & Rus, V. (2001, May). Transformation of wordNet glosses into logic forms. In *FLAIRS Conference Proceedings*, (pp. 459-463).. Retrieved from: <http://www.aaai.org/Papers/FLAIRS/2001/FLAIRS01-088.pdf>
- Mylopoulos, J., & Levesque, H. (1983). An overview of knowledge representation. In *GWAI-83* (pp. 143-157). Berlin-Heidelberg, Germany: Springer.
- Noy, N., & McGuinness, D. (2000). *Ontology development 101: a guide to creating your first ontology*. Retrieved from: http://liris.cnrs.fr/~amille/enseignements/Ecole_Centrale/What%20is%20an%20ontology%20and%20why%20we%20need%20it.htm
- Ono, K., Kawano, S., Fukazawa, Y., & Kadokura, T. (1992). A resolution method from predicate logic specification into executable code. In *Proceedings of the Twenty-Fifth Hawaii International Conference on System Sciences, 1992*. (Vol. 2, pp. 480-487). IEEE.
- Parnas, D. L. (1993). Predicate logic for software engineering. *Software Engineering, IEEE Transactions on*, 19(9), 856-862.
- Ramírez, S., Alonso, Y., Hernández, V., Arias, A., & La Rosa, D. (2010). Comparando UML y OWL en la representación del conocimiento: correspondencia sintáctica. *Revista Española de Innovación, Calidad e Ingeniería del Software*, 6(3), 84-94
- Repositorio Unilibre [web site] (2015). Retrieved from: <http://repository.unilivre.edu.co/handle/10901/1>
- Ruckhaus, E. (2005). *Lógicas descriptivas y ontologías*. Retrieved from: <http://ldc.usb.ve/~ruckhaus/materias/ci7453/clase51.pdf>
- Shi, Z., Liu, Z., & Chen, J. (2010). Using logic rules for concept refinement learning in first order logic. *2010 IEEE Fifth International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA)*, (pp. 444-448). IEEE.
- Sinab [web site] (2015). Retrieved from: <http://www.sinab.unal.edu.co/>
- Tayal, M., Raghuwansh, M., & Latesh, M. (2013). Knowledge representation: predicate logic implementation using sentence-type for Natural Languages. *Circuits, Power and Computing Technologies (ICCPCT)*, (pp. 1264-1269). IEEE.
- Teseo [web site] (2015). Retrieved from: <https://www.educacion.gob.es/teseo/irGestionarConsulta.do>
- Universidad del Rosario (2011). *Reglamento de trabajos de grado de la facultad de jurisprudencia*. Retrieved form: http://www.urosario.edu.co/jurisprudencia/pregrados/documentos/Reglamento_de_trabajos_de_grado_Jurisprudencia.pdf
- Wang, Y. (2010). Research on the construction of ontology-based criminology knowledge base. *2nd IEEE International Conference on Network Infrastructure and Digital Content*.
- Wang, Y. (2010). Research on the construction of ontology-based criminology knowledge base. In *2010 2nd IEEE International Conference on Network Infrastructure and Digital Content*, (pp. 123-128). IEEE.
- Yoo, S., & Park, C. H. (1993). An inference browser to verify knowledge bases. In *Fifth International Conference on Tools with Artificial Intelligence, 1993. TAI'93. Proceedings.*, (pp. 466-467). IEEE.

CURRICULUM VITAE

Germán Alejandro Pinilla García Student and researcher at the Universidad Distrital Francisco José de Caldas (Bogotá, Colombia). Miembro of Interoperabilidad Tecnológica y Semántica [INTECSE] research team / Estudiante investigador de la Universidad Distrital Francisco José de Caldas (Bogotá, Colombia). Miembro del Grupo de investigación Interoperabilidad Tecnológica y Semántica [INTECSE].

Julio Barón Velandia Professor and researcher at the Universidad Distrital Francisco José de Caldas (Bogotá, Colombia) / Profesor e investigador de la Universidad Distrital Francisco José de Caldas (Bogotá, Colombia).